

On the complexity of the Rank Syndrome Decoding problem

P. Gaborit¹, O. Ruatta¹ and J. Schrek¹

Université de Limoges, XLIM-DMI,
123, Av. Albert Thomas
87060 Limoges Cedex, France.
philippe.gaborit,julien.schrek,olivier.ruatta@unilim.fr

Abstract. In this paper we propose two new generic attacks on the Rank Syndrome Decoding (RSD) problem. Let C be a random $[n, k]$ rank code over $GF(q^m)$ and let $y = x + e$ be a received word such that $x \in C$ and the $\text{Rank}(e) = r$. The first attack is combinatorial and permits to recover an error e of rank weight r in $\min(O((n-k)^3 m^3 q^{\lceil \frac{km}{n} \rceil}), O((n-k)^3 m^3 q^{(r-1)\lceil \frac{(k+1)m}{n} \rceil}))$ operations on $GF(q)$. This attack dramatically improves on previous attack by introducing the length n of the code in the exponent of the complexity, which was not the case in previous generic attacks. The second attack is based on algebraic attacks: based on the theory of q -polynomials introduced by Ore we propose a new algebraic setting for the RSD problem that permits to consider equations and unknowns in the extension field $GF(q^m)$ rather than in $GF(q)$ as it is usually the case. We consider two approaches to solve the problem in this new setting. Linearization technics show that if $n \geq (k+1)(r+1) - 1$ the RSD problem can be solved in polynomial time, more generally we prove that if $\lceil \frac{(r+1)(k+1)-(n+1)}{r} \rceil \leq k$, the problem can be solved with an average complexity $O(r^3 k^3 q^{\lceil \frac{(r+1)(k+1)-(n+1)}{r} \rceil})$. We also consider solving with Gröbner bases for which we discuss theoretical complexity, we also consider hybrid solving with Gröbner bases on practical parameters. As an example of application we use our new attacks on all proposed recent cryptosystems which reparation the GPT cryptosystem, we break all examples of published proposed parameters, some parameters are broken in less than 1 s in certain cases.

Keys words: cryptanalysis, rank metric, algebraic attacks, Gröbner bases, coding theory

1 Introduction

There exist several alternative problems to classical cryptography based on number theory: besides lattice based cryptography and multivariate cryptography, code-based cryptography has been recently the object of papers [14,23,1,4] considering in details the practical complexity of the syndrome decoding problem for random codes for the Hamming metric. The rank metric for coding theory was introduced by Gabidulin in 1985 in [15] and he proposed a family of codes, the Gabidulin codes, analogous to Reed-Solomon codes in Hamming metric, which can be decoded in polynomial time. The Rank Syndrome Decoding (RSD) problem is the analogous for rank metric of the Syndrome Decoding problem for Hamming distance. Concerning cryptography, Gabidulin and *al.* proposed a few years later in [18] a cryptosystem (GPT) analogous to the McEliece cryptosystem but for rank metric. One of the advantage of rank metric is that the complexity of the best known attacks for solving the RSD problem have an exponential complexity which is quadratic in the parameters of the system. For C a $[n, k]$ code over $GF(q^m)$ that one wants to decode for an error of rank r , the 1996 attack by Chabaud and Stern [7] has an exponential term in $q^{(m-r)(r-1)}$ and the 2003 attack by Ourivski and Johansson [25] has an exponential term in $q^{(k+1)(r-1)}$. It means that in practice very high security in 2^{80} can potentially be obtained with a public key of only a few thousands bits for the generic RSD problem, when for Hamming distance for instance, relying on the generic Syndrome Decoding (SD) problem means considering matrices of at least several hundred thousands bits. Because of the strong structure of Gabidulin codes, the GPT cryptosystem has been the object of several structural attacks over the years and several variations [17] for hiding the structure of the Gabidulin codes, like the Rank Reducible codes, have been proposed, with always public keys size of order 10.000 bits. Besides the GPT system, Faure and Loidreau [13] proposed a cryptosystem also relying on the Gabidulin codes but

different from the GPT approach. At last public key zero-knowledge authentication schemes relying directly on random instances of RSD and with very small public keys have been proposed like [8] or very recently [19].

In 2005 Overbeck proposed a new structural attack [26,27] (see also the long version in J. of Crypto [28]), which permits to recover the structure of Gabidulin codes when hidden in different forms. His attack broke indeed all proposed parameters (at that time) of cryptosystems based on hiding the Gabidulin codes. A few years later, new parameters have been proposed [21,29] which resist the attack by Overbeck.

Meanwhile besides the Overbeck attack which is a structural attack only related to Gabidulin codes, the complexity of the generic RSD problem has not evolved for almost 10 years. In particular when looking at the exponential complexity of [7] and [25], it is striking that the exponential does not depend on the length the code. Besides these combinatorial attacks, an algebraic approach was also proposed in [20] but with limited results as soon as r was greater than 2 or 3, eventually the case $n = m$ is indirectly considered in [10]. Overall the RSD problem appears to be a cryptographic problem with a strong potential which seems under exploited.

Our contribution: In this paper we consider the complexity of solving the generic RSD problem we propose two new approaches, the first approach is combinatorial and generalizes a particular Hamming distance attack based on the error support in a rank metric context. Our attack can be seen as a generalization of both [7] and [25] and permits to include the length of the code in the exponential term of the complexity. For the second approach we introduce a new algebraic setting for solving the RSD problem, our setting relies on q -polynomials (or linearized polynomials) introduced by Ore and minimize the number of unknowns by giving an algebraic setting in the extension field $GF(q^m)$ rather than in $GF(q)$ as it is the case in general. We consider several ways to solve the problem in this setting: an hybrid generalization approach and a, hybrid solving with Gröbner bases. We apply our attack and break all reparation of the GPT cryptosystem proposed after the Overbeck attack. In practice for considered parameters algebraic attacks based on the new annihilator polynomial setting give the best results.

The paper is organized as follows: Section 2 recalls basic facts on rank codes , Section 3 explains the first attack based on error support, Section 4 introduces the new algebraic setting based on annihilator polynomials, Section 5 propose a solving of the setting with linearization, Section 6 considers solving with Gröbner basis and at last Section 7 deals with application of the attacks to specific cryptosystems parameters.

2 Background on rank metric, rank codes and algebraic systems

2.1 Definitions and notation

Notation :

Let q be a power of a prime p , m an integer and let V_n be a n dimensional vector space over the finite field $GF(q^m)$. Let $\beta = (\beta_1, \dots, \beta_m)$ be a basis of $GF(q^m)$ over $GF(q)$.

Let \mathcal{F}_i be the map from $GF(q^m)$ to $GF(q)$ where $\mathcal{F}_i(x)$ is the i -th coordinate of x in the basis β .

To any $v = (v_1, \dots, v_n)$ in V_n we associate the matrix $\bar{v} \in \mathcal{M}_{m,n}(GF(q))$ in which $\bar{v}_{i,j} = \mathcal{F}_i(v_j)$.

The rank weight of a vector v can be defined as the rank of the associated matrix \bar{v} . If we name this value $\text{rank}(v)$ we can have a distance between two vectors x, y using the formula $\text{rd}(x, y) = \text{rank}(x - y)$.

2.2 Codes for the rank distance

We refer to [22] for more details on codes for the rank distance.

A rank code C of length n and dimension k over $\text{GF}(q^m)$ is a subspace of dimension k of $\text{GF}(q^m)^n$ embedded with the rank metric. The minimum rank distance of the code C is the minimum rank of non-zero vectors of the code.

2.3 Rank distance and cryptography

The Syndrome Decoding problem for Hamming distance is written as:

Syndrome Decoding problem (SD)

Let H be a $((n-k) \times n)$ matrix over $\text{GF}(q^m)$ with $k \leq n$, $i \in \text{GF}(q^m)^k$ and ω an integer. The problem is to find s such that $wt(s) \leq \omega$ and $HS^t = i$ where wt denotes the Hamming weight.

The problem was proven NP-hard in [3] and is considered hard in general, especially when the matrix H is chosen at random. The best known algorithms for solving this problem are all exponential in ω , a recent survey on this complexity can be found in [14].

The previous problem can be naturally extended to the rank distance:

Rank Syndrome Decoding problem (RSD) Let H be a $((n-k) \times n)$ matrix over $\text{GF}(q^m)$ with $k \leq n$, $i \in \text{GF}(q^m)^k$ and r an integer. The problem is to find s such that $\text{rank}(s) = r$ and $HS^t = i$.

In that case it is not proven that the problem is NP-hard, but the relation with the Hamming case and the fact that the best known algorithms are all exponential makes this problem difficult in practice and the problem is generally believed to be hard.

There are two main approaches to this problem in the case of the rank matrix : Chabaud and Stern proposed an algorithm to solve the problem in $O((nr+m)^3 q^{(m-r)(r-1)})$ (see [7]) Ourivski and Johansson proposed two algorithms, the first one improves the polynomial part of the basis enumeration approach of [7] and is in $O((k+r)^3 q^{(m-r)(r-1)+2})$, the second uses a coordinates enumeration and is in $O((k+r)^3 r^3 q^{(r-1)(k+1)})$ (see [25]).

2.4 Polynomial solving

Some attacks proposed here consist to reduce the RSD problem to solving a polynomial system. Let us, now, introduce the problem of solving polynomial systems:

Problem: polynomial system solving (PoSSo):

Input: $f_1(x_1, \dots, x_u), \dots, f_t(x_1, \dots, x_u)$ polynomial over $\mathbb{K}[x_1, \dots, x_u]$ where \mathbb{K} is a field. **Goal:** Find all $\mathbf{z} = (z_1, \dots, z_u) \in \mathbb{K}^u$ such that $f_1(\mathbf{z}) = \dots = f_t(\mathbf{z}) = 0$.

We will use two main methods to solve this problem: linearization (when we have enough equations) and Gröbner bases (this a general approach). It is well known that PoSSo problem is NP-hard even if all the f_i are of degree 2 (in this case the problem is called \mathcal{MQ} for multivariate quadratic). Gröbner basis is a systematic tool to solve the PoSSo problem. When such a system has a finite number of solutions, it is said to be zero-dimensional. We will only consider zero-dimensional systems here since the roots coordinates are in a finite field and the field equations on each variable form a zero-dimensional system by itself.

3 Error support attack

3.1 Background on information set decoding for Hamming distance

The best algorithms for decoding general random codes for Hamming distance is the information set decoding approach [14]. This method can be considered in two different ways. Consider for instance G a generator matrix of a $[n, k]$ (binary) code and let H a parity check matrix of G .

The first original approach starts from the received word $y = xG + e$ and consists in guessing a set of k coordinates of y with no error (an information set), once such a set is found with a probability $\frac{\binom{n-t}{k}}{\binom{n}{k}}$, a linear inversion of a $k \times k$ matrix permits to recover x . This is what is done in some sense for rank codes by Ourivski and Johansson in [25].

Another approach for Hamming distance consists in starting from the syndrome $H.y^t$ of length $n - k$ of the received vector. The basic idea of the decoding algorithm consists in guessing a set of $n - k$ coordinates which contains the support of the error e , it can be obtained with probability $\frac{\binom{n-t}{n-k}}{\binom{n}{n-k}}$. Then since one gets $n - k$ equations from the syndrome equations and a set of $n - k$ coordinates containing the error support, it is possible to recover the error e by a $(n - k) \times (n - k)$ matrix inversion from the syndrome of the message.

It turns out that because of the properties of binomial coefficients, the two previous probabilities are equal and hence lead to the same exponential complexity for these two approaches (only in their simple form though - see recent improvements [14,23,1,4]). Meanwhile one can remark that, although these attacks are both considered as 'information set decoding', the second approach is not really connected with the notion of information set, but rather with the notion of error support.

We want to generalize the latter error support approach in the case of rank codes. We will see that at the difference of Hamming distance, for rank distance these two approaches lead to different exponential complexities and that the error support approach leads in general to a better complexity than the information set approach (corresponding to the Ourivski-Johansson approach).

3.2 General idea

Let C be a $[n, k]$ random code over $GF(q^m)$ with generator matrix G of size $k \times n$ and suppose one receives $y = c + e$ for $c \in C$ and $\text{rank}(e) = r$, in particular for $e = (e_1, \dots, e_n)$ there exists a subspace E of dimension r which contains all the error coordinates e_i . If one denotes by (E_1, \dots, E_r) a basis of E , one gets that: $\forall i, 1 \leq i \leq n$, there exists $e_{ij} \in GF(q)$ ($1 \leq i \leq n, 1 \leq j \leq r$) such that $e_i = \sum_{j=1}^r e_{ij} E_j$.

Let now H be a matrix of the dual code of C , then one gets

$$H.e^t = H.y^t. \quad (1)$$

In a context of rank distance the notion of support corresponds to the notion of error space E since E contains all possible coordinate errors. Notice that for rank distance the support is a notion related to value of the coordinate errors e_i , when for Hamming distance the notion concerns a set of coordinates.

Now we want to guess a support E' which contains the support E ; an important point is the fact that for such a support E' it has to be possible to recover the error e by solving a linear system (as for Hamming distance). In the case of rank distance, we have the rank syndrome equations. There are $n - k$ equations over the extension field $GF(q^m)$ given by the rank syndrome, when writing these equations over the small field $GF(q)$ we get $(n - k)m$ equations on the small field. Now suppose we know E' of dimension r' which contains E , then each error coordinate e_i can be written as an element

of E' . If we denote by $(E'_1, \dots, E'_{r'})$ a basis of E' in $GF(q^m)$ over $GF(q)$, then there exist $e'_{ij} \in GF(q)$ such that:

$$\forall i, 1 \leq i \leq n, \quad e_i = \sum_{j=1}^r e'_{ij} E'_j.$$

Since E' is fixed (and hence the E'_i), it gives $r' \cdot n$ unknowns (the e'_{ij}) in $GF(q)$ and hence it is possible to recover the errors coordinates e_i by solving a linear system as long as $r'n \leq (n - k)m$.

3.3 Error support attack

If one also uses the fact that there is a rank code structure, the previous idea permits to prove the following proposition:

Proposition 1. *Let C be a $[n, k]$ random code over $GF(q^m)$ with generator matrix G of size $k \times n$ and suppose one receives $y = c + e$ for $c \in C$ and $\text{rank}(e) = r$. Then one can recover c with an average complexity: $\min(O((n - k)^3 m^3 q^{r \lfloor \frac{km}{n} \rfloor}), O((n - k)^3 m^3 q^{(r-1) \lfloor \frac{(k+1)m}{n} \rfloor}))$.*

Proof. Let C be a $[n, k]$ random code over $GF(q^m)$ with generator matrix G of size $k \times n$ and suppose one receives $y = c + e$ for $c \in C$ and $\text{rank}(e) = r$, in particular there exists a subspace E of dimension r which contains all the errors coordinates e_i . Let now H be a matrix of the dual code of C , then one gets $H \cdot e^t = H \cdot y^t$. Suppose now one knows a subspace E' of dimension r' which contains E , then for all $e_i (1 \leq i \leq n)$, if we denote by $E'_1, \dots, E'_{r'}$ a basis of E' , there exist $e'_{ij} \in GF(q)$ such that:

$$e_i = \sum_{j=1}^{r'} e'_{ij} E'_j.$$

Equation (1) gives $(n - k)m$ equations over the small field, the number of unknowns derived from the e'_{ij} is $r'n$. Hence it is possible to recover the e'_{ij} (and therefore the e_i) by solving a linear system, as long as

$$r'n \leq (n - k)m,$$

and hence :

$$r' \leq \lfloor \frac{(n - k)m}{n} \rfloor$$

(for $\lfloor a \rfloor$: floor of a (the integer part of a)).

Let now be E' a subspace of dimension r' over $GF(q)$ of $GF(q^m)$, supposing that everything is random the probability that E of dimension r is included in E' of dimension r' (for $r' \geq r$) is $q^{-(m-r')r}$. Indeed consider a basis of E , one gets $E \subset E'$ if and only any element of a given basis of E is included in E' . Since any vector of a basis of E has a probability $\frac{q^{r'}}{q^m} = q^{-(m-r')}$ to be in E' (the number of element of E' divided by the number element in $GF(q^m)$), the probability that $E \subset E'$ is therefore $q^{-(m-r')r}$.

Hence if one takes $r' = \lfloor \frac{(n-k)m}{n} \rfloor = \lfloor m - \frac{km}{n} \rfloor$ one gets a probability that E is included in a random space E' of dimension r' , which is $q^{-(m-r')r} = q^{-r \lfloor \frac{km}{n} \rfloor}$. Hence if one also consider the complexity of the matrix inversion one gets the first proposed complexity of the proposition.

Now it is also possible to use the code structure and decrease the value of r by one, when increasing the value of k by one in the exponential coefficient, it is an interesting point since in practice, r is in general small and k is bigger than r .

The idea works as follows: one starts again, from the equation $y = xG + e$, we introduce a new $(k+1) \times n$ matrix G' obtained from G by adding a last row y . Now e belongs to the code C' generated by G' , but more generally since C' is a code over $GF(q^m)$, for any $\alpha \in GF(q^m)$, the vector αe is also in G' . The idea now is to fix a special value of α which will fix an element of the searched error space, it will decrease by 1 the number of basis element which are to be included in E' . Then once the space αE is recovered, one recovers the α and the original E .

To go in more detail on this idea: we suppose without loss of generality that $e_1 \neq 0$, if one considers the subspace $e_1^{-1}E$ it has still dimension r but contains the vector 1. One can apply the same method that previously but this time the code has dimension $k+1$ and one knows an element of E . The number of syndrome equations over $GF(q)$ is $(n-k-1)m$. And hence the dimension r' of E' must satisfy: $r' \leq \lfloor \frac{(n-k-1)m}{n} \rfloor$. Since one knows that $1 \in E$, one just need that the remaining $r-1$ elements of a basis of E are also in E' , which gives a probability $q^{-(r-1)\lfloor \frac{(k+1)m}{n} \rfloor}$. Once we recover $e_1^{-1}E$, taking e_1^{-1} as unknown in syndrome equations permits to recover it easily at almost no cost. Overall if one adds the polynomial complexity one gets the second complexity of the proposition. □

Remark 1: Comparison with previous attacks

In term of support, the basis enumeration attack corresponds to enumerate all possible supports of the error, it is the equivalent in Hamming distance, to enumerate all combination of error but with exact weight: the weight of the error. Such an approach does not take in account the fact that one knows $(n-k)$ linear equations in the extension field. Hence our attack can be seen as a combinatorial generalization of this point a view, in particular our attack is always better in term of exponential complexity. Our attack is also better in term of exponent complexity than [25] as soon as $n \geq m$, which is often the case in proposed parameters. Overall our attack can be seen as a generalization of the previous attacks [7].

remark 2: False solutions

There is the theoretical possibility that false solutions appear in the solving of the linear system, now since we consider the system as random, this case does not happen on the average. In practice with a strong probability we find only one solution to the system: the searched.

4 Annulator polynomial setting

We now consider a new algebraic setting, in order to do so we need to recall basic facts on q -polynomials.

4.1 Background on q -polynomials and annulator polynomials

We first recall some definitions on q -polynomials introduced by Ore in [24].

Definition 1. A q -polynomial of q -degree r in $GF(q^m)$ is a polynomial of the form:

$$P(x) = \sum_{i=0}^r p_i x^{q^i}, \quad \text{for } p_r \neq 0.$$

One can remark that since the application $x \rightarrow x^q$ is the Frobenius of $GF(q^m)/GF(q)$, any q -polynomial of q -degree r over $GF(q^m)$ can be seen as a linear application over $GF(q^m)$ considered as a vector space of dimension m over $GF(q)$.

In particular q -polynomials satisfy:

$$\forall x, y \in GF(q^m), \forall \alpha, \beta \in GF(q), \quad P(\alpha x + \beta y) = \alpha P(x) + \beta P(y).$$

In particular if a and b are roots of a q -polynomial P , then $P(a) = P(b) = 0 = P(a + b)$, and clearly the roots of a q -polynomial of q -degree r form a vector space over $GF(q)$ of dimension at most r .

The set of q -polynomials in $GF(q^m)$ has very nice properties, in particular it has a structure of non-commutative ring when it is embedded with the two following operations:

$$\text{Addition} : (P + Q)(x) = P(x) + Q(x).$$

$$\text{Composition} : (PoQ)(x) = P(Q(x)).$$

In [24], Ore describes how any r -dimensional subspace over $GF(q)$ in $GF(q^m)$ can be characterized as the set of root of a q -polynomial of q -degree r . He gives simple explicit polynomial constructions which permit in particular to construct such a polynomial from a given subspace. He proves the following proposition that we will use to define our algebraic setting:

Proposition 41 (Ore) *For any subspace E of $GF(q^m)$ over $GF(q)$ of dimension r there exists a unique monic q -polynomial P of q -degree r , such that:*

$$\forall z \in E, \quad P(z) = 0.$$

In the following an annihilator polynomial will be a q -polynomial which zeros are a given subspace of $GF(q^m)$. Such a polynomial annihilates in some sense the element of a given subspace of $GF(q^m)$.

4.2 A new algebraic setting for solving the rank decoding problem based on the annihilator polynomial

Let C be a $[n, k]$ random code over $GF(q^m)$ with generator matrix G of size $k \times n$. We denote by G_i the i^{th} rows of G and by g_{ij} the elements of G . Suppose one receives $y = c + e$ for $x \in C$ and $\text{rank}(e) = r$. Traditional algebraic settings for solving the rank distance problem [20] use in general as unknowns : r unknowns in $GF(q^m)$ for a basis of E , k unknowns in $GF(q^m)$ for the c_i and $n \times r$ unknowns in $GF(q)$ for the coordinates of the e_i in E . We now describe a new algebraic setting which has only $k + r$ unknowns in $GF(q^m)$.

The important point of this setting is given by the fact that the Annulator polynomial of Proposition 41 permits to characterize in an optimal way the notion that a matrix has a given rank, since all subspaces of rank r can be described as the set set of roots of a q -polynomials of q -degree r , hence simply by the r coefficients in $GF(q^m)$ of the q -polynomial.

Let $c = \sum_{i=1}^k c_i G_i$, $e = (e_1, \dots, e_n)$ and $y = (y_1, \dots, y_n)$. Since e has rank r , the subspace E generated by the e_i has dimension r . By Proposition 41 there exists a unique monic annihilator q -polynomial $P(x) = \sum_{i=0}^r p_i x^{q^i}$ with $p_r = 1$ such that $\forall z \in E, \quad P(z) = 0$. Hence we obtain:

$$\forall j, 1 \leq j \leq n, \quad P(y_j - \sum_{i=1}^k c_i g_{ij}) = P(e_j) = 0,$$

which gives n equations in the $k + r$ unknowns: $c_i (1 \leq i \leq k)$ and $p_j (0 \leq j \leq r - 1)$.

This new setting has unknowns has less unknowns than previous settings since all unknowns are in $GF(q^m)$, the general monomials of the system are of the form $p_j c_i^{q^j}$: they are quadratic terms in c_i and p_j , meanwhile the degree of the terms in c_i are exponential in q^r . Hence on one side we decrease the number of unknowns and on the other side we increase the degree of the equations.

We are now interested by the way to solve equations in this new setting, we will consider two ways: linearization and solving with Gröbner basis:

5 Solving by linearization

5.1 Basic approach

A basic approach consists in counting the number of different monomials in the c_i and the p_j and independent unknowns and the number of equations, in our setting, although the degree of equation is very high it turns out that the equations are also very sparse so that there are not so many different monomials, it is possible to obtain the following result:

Proposition 2. *Let C be a $[n, k]$ random code over $GF(q^m)$ with generator matrix G of size $k \times n$ and suppose one receives $y = c + e$ for $c \in C$ and $\text{rank}(e) = r$. If $n \geq (r+1)(k+1) - 1$ the complexity of solving the rank decoding problem is polynomial in $((r+1)(k+1) - 1)^3$ operations in $GF(q^m)$.*

Proof. We saw in previous section how the new setting could be described: Let $c = \sum_{i=1}^k c_i G_i$, $e = (e_1, \dots, e_n)$ and $y = (y_1, \dots, y_n)$. Since e has rank r , the subspace E generated by the e_i has dimension r . By Proposition 41 there exists a unique monic annihilator q -polynomial $P(x) = \sum_{i=0}^r p_i x^{q^i}$ with $p_r = 1$ such that $\forall z \in E, P(z) = 0$. Hence we obtain:

$$\forall j, 1 \leq j \leq n, \quad P(y_j - \sum_{i=1}^k c_i g_{ij}) = P(e_j) = 0, \quad (2)$$

which gives n equations in the $k + r$ unknowns: $c_i (1 \leq i \leq k)$ and $p_j (0 \leq j \leq r - 1)$. Now the system we obtain is quadratic in the unknowns c_i and p_i . Such a non linear system can be solved through Gröbner basis, but it is also possible to solve by linearization, indeed in this case by linearization we obtain $(r+1)(k+1) - 1$ terms:

- $k \cdot r$ terms of the form : $p_j c_i^{q^j}$ for $1 \leq i \leq k$ and $0 \leq j \leq r - 1$
- k terms of the form: $c_i^{q^r}$ for $1 \leq i \leq k$ (corresponding to the term $p_r = 1$).
- r terms of the form : p_j for $0 \leq j \leq r - 1$ (corresponding to the scalar coordinates of y)

Hence overall $(r+1)(k+1) - 1$ linearized terms. In the case where the number of equations n satisfy $n \geq (r+1)(k+1) - 1$, the problem can hence be solved on the average by solving a linear system over $GF(q^m)$ with $(r+1)(k+1) - 1$ unknowns.

□

5.2 An hybrid advanced approach

We saw how it was possible depending on conditions on n, k and r to solve directly the problem, now what happens if such a condition is not fulfilled. We saw that in basic linearization of previous section that the number of unknowns was quadratic in r and k . It is possible to decrease this number by guessing an error. Suppose indeed that an error e_j is zero, recall that:

$$\forall j, 1 \leq j \leq n, \quad y_j = \sum_{i=1}^k c_i g_{ij} + e_j,$$

then if $e_j = 0$ one obtains a linear equation in the c_i , which permits to substitute one of the c_i by a linear combination of the others in all rows equations of the code.

In particular it means if one can find an error $e_j = 0$, then one can decrease the number of c_i by one and hence decrease the number of unknowns in the linearization by $(r+1)$ terms. Now since the error $e_i \in E$ of dimension r , for random e_i the probability that $e_i = 0$ is q^{-r} .

This idea is precised in :

Proposition 3. *Let C be a $[n, k]$ random code over $GF(q^m)$ with generator matrix G of size $k \times n$ and suppose one receives $y = c + e$ for $c \in C$ and $\text{rank}(e) = r$. If there exists an integer $t \leq k$ such that $n - t \geq (r + 1)(k + 1 - t) - 1$ then complexity of solving the rank decoding problem has an average complexity bounded above by $O((nkt + r^3 k^3)q^{rt})$ operations in $GF(q^m)$.*

Proof. Our algebraic setting gives n equations in c_i and p_j . Suppose that we know that for a given equation, the error e_j is zero, then we obtain a linear equation $(\sum_{i=1}^k c_i g_{ij})$ with only unknowns the c_i . Suppose that c_1 (for instance) is written in terms of the others $c_j (c_j \neq 1)$, then substituting c_1 by a linear equation in the $c_j (c_j \neq 1)$, in all the n equations given by the relation $y = cG + e$ gives a new system of equations, with $n - 1$ linear equations without c_1 and one equation that is kept aside with c_1 . Since the error rank is still the same, one knows that the annihilator polynomial is still an annihilator polynomial since the remaining errors e_j are the same. Hence one can still use equation (2) of the previous section, but this time the number of unknowns c_i has decreased by one. We hence obtain a new linearized system of equations with only $(r + 1)(k + 1) - 1 - (r + 1)$ terms. Now since we have used an equation to describe c_1 from the c_i we have one equation less (which contain terms with c_1) and hence $n - 1$ independent equations without c_1 .

We hence saw how to it was possible to decrease the number of linearized terms when a zero error e_j was known. Now all rows of the code permits to derive linear equations:

$$\forall i, 1 \leq i \leq n, \quad y_i = \sum_{j=1}^k c_j g_{ij} + e_i.$$

If one considers these n equations and consider new equations obtained by additions of multiplication of these equations by a random non-zero element of $GF(q)$, the obtained equations are still linear in c_j and since multiplication by an element of $GF(q)$ does not change the error support, the error obtained in the new equation can be considered as a random random element of E . Therefore we can deduce that the probability to obtain a zero error in the linear combination of these equations is $\frac{1}{q^r}$ since there are only q^r possible errors.

Repeating the process t times permits each time to decrease the number of linearized terms by $r + 1$ and reduces the number of equations to be used (ie: without the substituted c_i) by 1, with a probability of success of $\frac{1}{q^{rt}}$. The complexity of the attack has a probability part in q^{rt} and a polynomial part. The polynomial part consists in searching new equations with error zero, this part is negligible since one can use a method where one modifies very few equations for each new trial. Once a potential zero is found, after finding i zeros, one has to write the c_j in terms of c_l for $1 \leq l \leq j - 1$, and then modify the terms of each c_l with the terms coming from c_k . After t trials the cost is hence $\sum_{i=1}^t (k - i)(n - i)$. Then the last part is the solving of a linear system in $GF(q^m)$ with $(r + 1)(k + 1 - t) - 1$ unknowns. The overall polynomial cost is hence $\sum_{i=1}^t (k - i)(n - i) + ((r + 1)(k + 1 - t) - 1)^3$ operations in $GF(q^m)$. The first term can be bounded above by nkt and the second by $(r + 1)^3(k + 1)^3$, which gives the result.

□

Corollary 1. *Let C be a $[n, k]$ random code over $GF(q^m)$, suppose one receives $y = c + e$ for $c \in C$ and $\text{rank}(e) = r$. Then if $\lceil \frac{(r+1)(k+1)-(n+1)}{r} \rceil \leq k$, the error e can be recovered with complexity $O(r^3 k^3 q^{r \lceil \frac{(r+1)(k+1)-(n+1)}{r} \rceil})$.*

Proof. We apply the previous proposition: the condition $t \leq k$ such that $n - t \geq (r + 1)(k + 1 - t) - 1$ gives $t = \lceil \frac{(r+1)(k+1)-(n+1)}{r} \rceil$. In the complexity since in general for practical parameters $t \ll n$ we neglect the part in nkt . □

6 Solving with Gröbner basis

6.1 Solving polynomial systems: Gröbner basis approach

The notion of Gröbner basis is linked to the one of monomial term ordering. A monomial order is an order on monomials which is compatible with the product in order to have a pseudo-division with respect to such an order. Roughly speaking, a Gröbner basis $\mathcal{G} = \{g_1, \dots, g_s\}$ of the ideal generated by a set of polynomials f_1, \dots, f_n is a family such that, for each $h \in \mathbb{K}[x_1, \dots, x_u]$, then remainder of the pseudo-division of h with respect to \mathcal{G} is 0 if and only if h lies in the ideal (f_1, \dots, f_n) .

The lexicographical orders are particularly interesting since the shape of the Gröbner basis for such an order is the following:

$$g_1(x_1), g_{2,1}(x_1, x_2), \dots, g_{2,l_2}(x_1, x_2), \dots, g_{u,1}(x_1, \dots, x_u), \dots, g_{u,l_u}(x_1, \dots, x_u)$$

This structure allows to solve the original system. It was the initial motivation to the research of more efficient algorithms to compute Gröbner bases. Generally, computing a Gröbner basis for a lexicographical order is harder than computing one for graded ordering. But once you know a Gröbner basis for a graded order you can use the FGLM algorithm to have one for lexicographical order or use a solver that uses directly the structure of the pseudo-division by Gröbner basis with respect to graded order.

The more efficient algorithm to compute Gröbner basis is the F_5 algorithm of Faugère [9], but for experiments realized in this work had been made using the F_4 algorithm in MAGMA [6]. Here, we give a complexity result using the F_5 algorithm even if we use the F_4 algorithm since the use of F_5 algorithm has been carefully studied for cryptography. An important quantity for Gröbner basis computation of an ideal is the regularity of the generating system, denoted d_{reg} , defining the ideal. The number d_{reg} is the biggest degree reached in the Gröbner basis computation by the F_5 algorithm. In [12], the authors give a way to bound the complexity of the algorithm with respect to the regularity of the system:

Proposition 4. *The complexity of computing Gröbner basis of a zero-dimensional system of t equations in u variables using the F_5 algorithm is:*

$$\mathcal{O} \left(n * \binom{u + d_{reg} - 1}{d_{reg}}^\omega \right)$$

where d_{reg} is the degree of regularity of the system and $2 \leq \omega \leq 3$ is the linear algebra constant.

6.2 Gröbner bases for RSD

We will now use this technical background to study the original system, denoting $\mathbf{p} = (p_0, \dots, p_{r-1})$ and $\mathbf{c} = (c_1, \dots, c_k)$:

$$\forall i \in \{1, \dots, n\}, l_i(\mathbf{p}, \mathbf{c}) = \sum_{a=0}^r \left[(p_a y^{q^a}) - \left(\sum_{j=0}^k p_a c_j^{q^a} g_{i,j}^{q^a} \right) \right]. \quad (1)$$

Complexity issues of our method: The use of Gröbner bases is very important when $n < (r+1)(k+1) - 1$, possibly combined with guessing of some variables for an hybrid approach. Here, instead of $(r+1)(k+1)$ variables of the linear attack, we have $r(k+1)$ variables since we can assume that the polynomial is unitary ($p_r=1$).

The system is sparse and has a suitable structure. Even if it has a very algebraic definition, the notion of regularity is actually related to the one of randomness. The system is a semi-regular system. To

see this, first remark that this ideal (l_1, \dots, l_n) is proper since the system always has a solution. The other property to check is a consequence that the system inherits of the randomness of the underling code. The leading term of each l_i is provided by the term $\sum_{a=0}^r \sum_{j=0}^k p_a c_j^{q^a} g_{i,j}^{q^a}$ and so the leading term is

issue of $\sum_{j=0}^k c_j^{q^r} g_{i,j}^{q^r}$. This is a random homogeneous system of degree q^r since the coefficients come from the random matrix G . This insure us the good “random” behavior of the system.

We denote by $M_d(u)$ the set of monomial of degree d in u variables, we have $\#M_d(u) = \binom{u+d-1}{d}$. Following [9], the complexity to compute a Gröbner basis of an ideal of degree if regularity d_{reg} in a ring of polynomial of u variables with the F_5 algorithm is $\mathcal{O}((\#M_{d_{reg}}(u))^\omega)$.

Remark that all the equations have degree $q^r + 1$ in a way that d_{reg} is the first non positive coefficient of $\frac{(1-z^{q^r+1})^{rk}}{(1-z)^n}$. Since:

$$\frac{(1 - z^{q^r+1})}{(1 - z)} = \sum_{i=0}^{q^r} z^i,$$

we have that d_{reg} is the first non positive coefficient of:

$$\left(\sum_{i=0}^{q^r} z^i \right) (1 - z)^{kr-n}.$$

We obtain a complexity in $\mathcal{O}\left(n \binom{(k+1)r+d_{reg}}{d_{reg}}\right)$. We used the package describe in [5] to compute the regularity of some problems (and one can deduce a close formula for d_{reg} from the above computation). The sparseness of the system make the theoretical complexity evaluation to far from practical achievement. For instance, for the case where $q = 2^{24}$, $k = 12$, $r = 6$ (i.e. equations have degree $2^6 + 1$) and $n = 64$, we have a regularity of 200 and a complexity bound by 2^{152} ! But the running time to solve using F_4 in MAGMA is only few hours (and we can take advantage of the hybrid approach in order to improve the approach). It appears, experimentally, that the equations appearing in the computations are very very sparse and remain sparse. When the number of equations decrease, the algorithm destroy fast the sparse structure. So, the theoretical bound has generally no meaning by itself, but it reveals some structural properties of the formulation. Experimentally, the running time of the algorithm behaves as if we replace the degrees of the equations by their q -degree (here the degree is q^r and the q -degree is r). In the previous example, instead of a complexity of 2^{152} with the degree, the complexity with q -degree is 2^{55} and the algorithm effectively run within few hours. This remark is always valid in example as long as $n > r(k+1)$. This give a range of parameters for which the Gröbner bases approach improve the linearization. Furthermore, the hybrid approach extend naturally the advance approach as we will see below.

Comparison with other approaches: Other approaches, introduced in the context of cryptanalysis of systems based on MinRank problem can be extended to the RSD problem and reduce the considered problems to PoSSo problem just as we did in the previous paragraphs. We will show that our approach is of particular interest compare to those ones. The two methods has in common to get back to the linear algebra formulation and so, they work on the field $GF(q)$. We do not introduce here MinRank problem, we only adapt the attack to RSD. To do this, we use the reduction introduced in [10] to transform in poly-time a rank decoding problem to a MinRank problem. We also use the bounds given in [10] since, those authors give finer result in [12] and [11], but algorithm in [12] works only for square matrices (which is generally not true in our cases) and in [11] the algorithm is probabilistic and the complexity is not improved drastically. Using the reduction of [10], we reduce a RSD problem on $GF(q^m)$ with parameters k for the dimension of the code, n for it size and r for the error rank to a MinRang problem

of parameters m (number of rows of the matrices), n (number of column of the matrices), r (rank) and km (number of matrices). The method was develop only for square matrices, but it is possible to extend it to rectangular matrices. Using the theoretical bound of [12], the Kipnis-Shamir approach apply to a RSD problem of parameters n, k, r leads, when the generated MinRank problem is square, to an algorithm with complexity $\mathcal{O}\left(\binom{k*m+r(n-r)+d_{reg}}{d_{reg}}^\omega\right)$ (ω still denotes the linear algebra constant) and with $d_{reg} \leq 1 + \min\{k*m, (n-r)r\}$. Here, the number of variables depend of n and m in contrary to our approach and is the bound seem also very pessimistic. Finally, the minors approach apply to a RSD problem of parameters n, k, r needs $\mathcal{O}\left(\binom{k*m+r(n-r)+1}{r(n-r)+1}\right)$ operations over $GF(q)$ with some more restrictive conditions. So, even if it is possible to give bounds for this approach, the number of variable highly depend on the dimension m of the field $GF(q^m)$ over $GF(q)$ and of the number of equation n in the exponent. Our approach, staying in $GF(q^m)$, avoid the parameter m in the combinatorial factor (it is on the constant for the complexity of the basic operations on $GF(q^m)$). Furthermore, for our approach, the number of equation n does not appear in the combinatorial factor, but only on the regularity making regularity decrease when n rises.

6.3 Hybrid approach

Just as for the advance linearization attack, it is possible to make an hybrid approach making some guess on the values of some variables c_i . Since the number of variables for the Gröbner bases approach is $(k+1)*r$ each time we find a c_i , we reduce the number of variables of r . Furthermore, we reduce the number of variable without decreasing the number of equation a lot. It is to say that making guess on several variables improve the ratio of the number of equation over the number of variables. It is known that it is easier to compute an Gröbner basis of a very over-constrained non-coherent systems. We use this in order to define an heuristic: try to guess sufficiently many c_i to be able to check fast that the generated system is not coherent. There is tradeoff between the number t of c_i we try to guess (it gives a q^t factor to the complexity and decrease the probability of success) and the speed of checking if the system is not coherent.

7 Cryptanalysis of some cryptosystems

7.1 The GPT rank-based cryptosystem

The GPT cryptosystem is similar to the McEliece cryptosystem but works for rank distance. The Gabidulin codes are the equivalent of the Reed-Solomon codes for rank metric. The main problem in the cryptosystem consists in finding a way to hide the decoding matrix. For Hamming distance it is done through a permutation matrix. In the case of Gabidulin codes, several approaches have been proposed by adding words of small rank, by adding a scrambling matrix, introducing a new class of codes: the Rank reducible codes etc... All these systems lead to interesting parameters. There are two ways to attack such systems: a first way is structural and the attacker tries to recover the mask (or the hiding procedure) from the public key, based on the structural properties of the Gabidulin codes. In 2005 Overbeck [26] proposed a structural attack which broke many proposed parameters. After this attack some new parameters have been proposed which resist to this attack. We show in the following that these parameters are not secure either, meanwhile at the difference of Overbeck's attack, our attack is not structural but completely generic and depends only on code parameters.

7.2 Cryptanalysis of some proposed parameters in rank metric

In the following we apply our method on different reparation of GPT cryptosystem. Since the Basis enumeration and the Ourivski-Johansson attack were well known people proposed new variations

which focused on resisting to Overbeck attack, since in general it was rather easy to resist to the Basis and Coordinate enumeration attacks.

Several approaches have been proposed to resist Overbeck's attack [16,21,29,30,31], but only two papers propose published parameters: an approach by Loidreau in [21] and an 'advanced standard approach' by Gabidulin, Rashwan and Honory ([29,31]). In the following we show that all the proposed parameters in these papers are completely broken and can be practically recovered, even in less than 1s sometimes.

In the following we attack the RSD problem for $[n, k]$ codes for an error of rank r , $q = 2$ and an extension of size 2^m . In the following tables we give the different complexity of the different attack regarding the code used. Notice that our attacks are not structural attacks since we do not use any particular structure of the code. In the tables: 'OJ1' stands for the improved basis enumeration by Ouriski and Joahsson, 'OJ2' stands for coordinates enumeration, 'Over' stands for the complexity of the Overbeck attack, 'ES' stands for the complexity of the Error Support attack of Section 3, 'L' stands for the attack by linearization of section 5 (usually it does not work and hence we put ∞), 'LH' stands for the complexity of the attack by hybrid linearization when guessing zero coordinates errors and 'HGb' is the complexity (usually in time) when one attacks with hybrid solving with Gröbner basis in our new setting. We did not put the complexity with simple Gröbner basis since it usually does not finish. All our computations were done with the F4 version of Magma on a double core of a 2GHz INTEL with 8 Go RAM.

We now consider different type of reparation.

• Loidreau reparation [21]

The idea of the reparation is to add sufficiently many columns so that the Overbeck attack does not work. The author focus on the complexity of the Overbeck attack since there is no difficulty to resist other attacks since in previous complexity, the length n of the code did not appeared in the exponential complexity of the attack. The author starts from a [24, 12, 12] Gabidulin code which can correct 6 errors and proposes two sets of parameters for which he adds 40 random columns or 52 random columns. The following table gives the different complexities for our attacks.

Code parameters (n, k, r, m)	OJ1	OJ2	Over	ES	L	LH	HGb
(64, 12, 6, 24)	2^{104}	2^{85}	2^{80}	2^{50}	∞	2^{48}	2 hours
(76, 12, 6, 24)	2^{104}	2^{85}	2^{80}	2^{49}	∞	2^{36}	< 1 s

For hybrid Gröbner basis attack we add mix multiplied (by a non zero element of $GF(q)$) columns and fix 3 coordinates, that we hope to have error coordinate zero. We then construct our algebraic setting that we solve with Gröbner bases and the F4 algorithm. If the guessing was wrong a failure was obtained with F4 in an average of 0.13 s, repeating the process in a 4 processor computer permitted us to retrieve the solution in 2h. We also run the LH attack which in practice had a complexity in 2^{44} field operations, we run the attack in Magma and overall the HGb attack was far more efficient and faster than the attack with Gröbner bases. Our attacks show that all parameters sets proposed in [21] are completely broken, the second set of parameters which was supposed to be stronger than the first one can in fact be attacked in a few seconds with hybrid Gröbner bases attack.

• Cryptanalysis of Gabidulin et al. reparations [29,31]

In [29] and [31], Gabidulin et al. propose an approach and parameters (claimed with security 2^{80}) to resist Overbeck's attack, the approach called 'advanced approach for standard variant' proceeds by hiding as usual the generator matrix G with a matrix M with a special form, overall the proposed parameters can be attacked directly in decoding an error e of rank r in a $[n, k]$ code over $GF(2^m)$.

We give in the following table the different parameters proposed and the complexity , the two first parameters are from [29] and the two last ones are from [31] corresponding to a public key of size 4000b.

Code parameters (n, k, r, m)	Over	ES	L	LH	HGb
(28, 14, 3, 28)	2^{80}	2^{55}	∞	2^{49}	2 days
(28, 14, 4, 28)	2^{80}	2^{70}	∞	2^{65}	not finished
(20, 10, 4, 20)	2^{80}	2^{56}	∞	2^{51}	5 days
(20, 12, 4, 20)	2^{80}	2^{60}	∞	2^{60}	not finished

Experimental results show that it was possible to recover the message in 2 and 5 days with an hybrid Gröbner bases attack for the first and third set of parameters. In particular it shows that parameters proposed in [31] with a public key of 4000b are clearly unsafe. For the second and fourth case the computation could not finish with hybrid Gröbner bases meanwhile the hybrid linearization attack (without Gröbner bases) gives attack complexities of order 2^{60} which implies that these parameters can also be considered broken. Practical computation were done which shows that in practice the time estimation of the complexity followed these complexities.

8 Conclusion

In this paper we propose two new generic approaches to attack the RSD problem, both approaches have their own interest depending of the type of parameters considered. The first approach is combinatorial and improves considerably previous attacks and in particular permits to take account of the length of the code, which not the case previously. We also propose a new algebraic setting based on q -polynomials which permits to preserve the mathematical structure over the extension field, which is not the case in previous algebraic setting. At last we break all published parameters proposed to repair the GPT cryptosystem after Overbeck's attack. In practice the algebraic attacks do not work necessarily for all type of parameters but these attacks were more efficient than the first generic combinatorial attack on the parameters we attacked. The RSD problem seems still promising but still more work has to be done like for Hamming distance in order to have a clear view of the computational complexity of the problem. A future direction of work is to consider other type of cryptosystem, moreover it is an open question to try to generalize our combinatorial approach in order to apply the same type of ideas than for codes with Hamming distance [1,23,4,14].

References

1. Anja Becker, Antoine Joux, Alexander May, Alexander Meurer: Decoding Random Binary Linear Codes in $2^{n/20}$: How $1 + 1 = 0$ Improves Information Set Decoding. EUROCRYPT 2012: 520-536
2. Thierry P. Berger, Pierre Loidreau: Designing an Efficient and Secure Public-Key Cryptosystem Based on Reducible Rank Codes. INDOCRYPT 2004: 218-229
3. Berlekamp, E. and McEliece, R. and van Tilborg, H., On the inherent intractability of certain coding problems , IEEE Transactions on Information Theory, p. 384-386 (1978).
4. Daniel J. Bernstein, Tanja Lange, Christiane Peters: Smaller Decoding Exponents: Ball-Collision Decoding. CRYPTO 2011: 743-760
5. L. Bettale, J.-C. Faugère, and L. Perret. Hybrid approach for solving multivariate systems over finite fields. *Journal of Mathematical Cryptology*, 3(3):177-197, 2010.
6. W. Bosma, J. Cannon and C. Playoust The Magma algebra system. I. The user language *Journal of Symbolic Computation*, vol. 24, 3-4:235-265, 1997.
7. Florent Chabaud, Jacques Stern: The Cryptographic Security of the Syndrome Decoding Problem for Rank Distance Codes. ASIACRYPT 1996: 368-381
8. K. Chen , A New Identification Algorithm, in Cryptography: Policy and Algorithms, pp. 244-249, (1995)
9. J.-C. Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). *ISSAC'02: In Proceedings of the 2002 international symposium on Symbolic and algebraic computation*, pages 75-83, New York, NY, USA, 2002. ACM.

10. J.-C. Faugère, F. Levy-dit-Vehel, L. Perret. Cryptanalysis of MinRank. In *CRYPTO 2008*, LNCS 5157, pages 280–296. Springer Verlag, 2008.
11. J.-C. Faugère, M. Safey El Din, and P.-J. Spaenlehauer On the Complexity of the Generalized MinRank Problem. <http://arxiv.org/pdf/1112.4411.pdf>
12. J.-C. Faugère, M. Safey El Din, and P.-J. Spaenlehauer. Computing Loci of Rank Defects of Linear Matrices using Gröbner Bases and Applications to Cryptology. *ISSAC '10: Proceedings of the 2010 international symposium on Symbolic and algebraic computation*, pages 257–264, New York, NY, USA, 2010. ACM.
13. Cédric Faure, Pierre Loidreau: A New Public-Key Cryptosystem Based on the Problem of Reconstructing p-Polynomials. WCC 2005: 304–315
14. Matthieu Finiasz, Nicolas Sendrier: Security Bounds for the Design of Code-Based Cryptosystems. ASIACRYPT 2009: 88–105
15. Ernst M. Gabidulin, Theory of Codes with Maximum Rank Distance, Probl. Peredachi Inf, (21), pp. 3–16 (1985).
16. Ernst M. Gabidulin: Attacks and counter-attacks on the GPT public key cryptosystem. Des. Codes Cryptography 48(2): 171–177 (2008)
17. Ernst M. Gabidulin, Alexei V. Ourivski, Bahram Honary, Bassem Ammar: Reducible rank codes and their applications to cryptography. IEEE Transactions on Information Theory 49(12): 3289–3293 (2003)
18. Ernst M. Gabidulin, A. V. Paramonov, O. V. Tretjakov: Ideals over a Non-Commutative Ring and thier Applications in Cryptology. EUROCRYPT 1991: 482–489
19. Philippe Gaborit, Julien Schrek, Gilles Zémor: Full Cryptanalysis of the Chen Identification Protocol. PQCrypto 2011: 35–50
20. F. Levy-dit-Vehel and L. Perret, Algebraic decoding of rank metric codes, proceedings of YACC06.
21. Pierre Loidreau: Designing a Rank Metric Based McEliece Cryptosystem. PQCrypto 2010: 142–152
22. P. Loidreau, Properties of codes in rank metric, <http://arxiv.org/abs/cs/0610057>
23. Alexander May, Alexander Meurer, Enrico Thomae: Decoding Random Linear Codes in $\tilde{O}(2^{0.054n})$. ASIACRYPT 2011: 107–124
24. O. Ore, On a special class of polynomials, Trans. American Math. Soc. (1933)
25. Ourivski, A. V. and Johansson, T., New Technique for Decoding Codes in the Rank Metric and Its Cryptography Applications, Probl. Inf. Transm.(38), 237–246 (2002)
26. Raphael Overbeck: Extending Gibson’s Attacks on the GPT Cryptosystem. WCC 2005: 178–188
27. Raphael Overbeck: Extending Gibson’s Attacks on the GPT Cryptosystem. WCC 2005: 178–188
28. Raphael Overbeck: Structural Attacks for Public Key Cryptosystems based on Gabidulin Codes. J. Cryptology 21(2): 280–301 (2008)
29. Haitham Rashwan, Bahram Honary, Ernst M. Gabidulin: On improving security of GPT cryptosystems. ISIT 2009: 1110–1114
30. Haitham Rashwan, Ernst M. Gabidulin, Bahram Honary: A Smart approach for GPT cryptosystem based on rank codes. ISIT 2010: 2463–2467
31. Haitham Rashwan, Ernst M. Gabidulin, Bahram Honary: Security of the GPT cryptosystem and its applications to cryptography. Security and Communication Networks 4(8): 937–946 (2011)